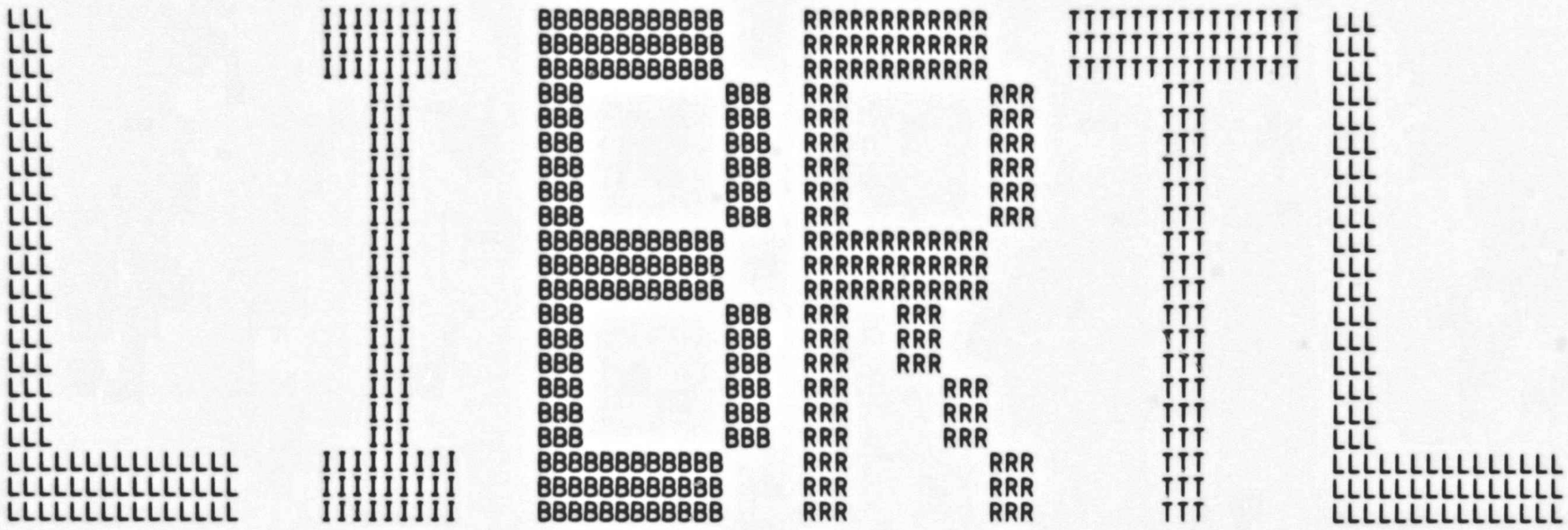
LLL IIIIIIIII BBBBBBBBBBB RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
LLL IIIIIIIII BBBBBBBBBBBB RRRRRRRRRRRR TTTTTTTTTTTTTTT LLL
LLL IIIIIIIII BBBBBBBBBBBB RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBBBBBBBBBB RRRRRRRRRRR TTT LLL
LLL III BBBBBBBBBBBB RRRRRRRRRRRR TTT LLL
LLL III BBBBBBBBBBBB RRRRRRRRRRRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLL III BBB BBB RRR RRR TTT LLL
LLLLLLLLLLLLLLL IIIIIIIII BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL IIIIIIIII BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL IIIIIIIII BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL

FILEID**LIBINITIA

```
LL          IIIIII   BBBBBBBB     IIIIII   NN      NN   IIIIII   TTTTTTTTTT   IIIIII     AAAAAA
LL          IIIIII   BBBBBBBB     IIIIII   NN      NN   IIIIII   TTTTTTTTTT   IIIIII     AAAAAA
LL            II     BB      BB     II     NN      NN     II         TT         II      AA      AA
LL            II     BB      BB     II     NN      NN     II         TT         II      AA      AA
LL            II     BB      BB     II     NNNN    NN     II         TT         II      AA      AA
LL            II     BB      BB     II     NNNN    NN     II         TT         II      AA      AA
LL            II     BBBBBBBB       II     NN  NN  NN     II         TT         II      AA      AA
LL            II     BBBBBBBB       II     NN  NN  NN     II         TT         II      AAAAAAAAAA
LL            II     BB      BB     II     NN    NNNN     II         TT         II      AAAAAAAAAA
LL            II     BB      BB     II     NN    NNNN     II         TT         II      AA      AA
LL            II     BB      BB     II     NN      NN     II         TT         II      AA      AA
LLLLLLLLLL  IIIIII   BBBBBBBB     IIIIII   NN      NN   IIIIII       TT       IIIIII    AA      AA
LLLLLLLLLL  IIIIII   BBBBBBBB     IIIIII   NN      NN   IIIIII       TT       IIIIII    AA      AA

LL          IIIIII     SSSSSSSS
LL          IIIIII     SSSSSSSS
LL            II     SS
LL            II     SS
LL            II     SS
LL            II     SS
LL            II       SSSSSS
LL            II       SSSSSS
LL            II             SS
LL            II             SS
LL            II             SS
LL            II             SS
LLLLLLLLLL  IIIIII   SSSSSSSS
LLLLLLLLLL  IIIIII   SSSSSSSS
```

Table of contents

(2) 57 HISTORY ; Detailed Current Edit History
(3) 78 DECLARATIONS
(4) 124 LIBSINITIALIZE - Library initialization procedure dispatcher

```
0000     1          .TITLE  LIB$INITIALIZE          ; Library initialization dispatcher
0000     2          .IDENT  /1-004/                 ; File: LIBINITIA.MAR  Edit: SBL1004
0000     3
0000     4 ;
0000     5 ;*****************************************************************************
0000     6 ;*                                                                           *
0000     7 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     8 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9 ;*  ALL RIGHTS RESERVED.                                                     *
0000    10 ;*                                                                           *
0000    11 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    12 ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    13 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    14 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    15 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    16 ;*  TRANSFERRED.                                                             *
0000    17 ;*                                                                           *
0000    18 ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    19 ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    20 ;*  CORPORATION.                                                             *
0000    21 ;*                                                                           *
0000    22 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    23 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    24 ;*                                                                           *
0000    25 ;*                                                                           *
0000    26 ;*****************************************************************************
0000    27 ;
0000    28 ;++
0000    29 ; FACILITY: RUN-TIME UTILITY PROCEDURE LIBRARY
0000    30 ;
0000    31 ; ENVIRONMENT:  User mode - not re-entrant.  Called only once
0000    32 ;                per image activation.
0000    33 ;
0000    34 ;
0000    35 ; ABSTRACT:
0000    36 ;
0000    37 ;       LIB$INITIALIZE is a software architecture procedure
0000    38 ;       which permits any library or user modular procedure to specify an initializa
0000    39 ;       to be called before the main program is called.
0000    40 ;       The initialization procedure can perform any once per-image
0000    41 ;       activation initialization which cannot be performed by the usual
0000    42 ;       compile-time and link-time static data initialization.
0000    43 ;
0000    44 ;--
0000    45 ;
0000    46 ; VERSION: 0
0000    47 ;
0000    48 ; HISTORY:
0000    49 ;
0000    50 ; AUTHOR:
0000    51 ;       Thomas N. Hastings, 6-Aug-77: Version 0
0000    52 ;
0000    53 ; MODIFIED BY:
0000    54 ;
0000    55 ;
```

```
0000     57          .SBTTL  HISTORY         ; Detailed Current Edit History
0000     58
0000     59
0000     60 ; Edit History for Version 0
0000     61 ;
0000     62
0000     63 ; 0-01  - initial version TNH 12-SEP-77
0000     64 ; 0-4   - Declare PSECT LIB$CODE SHR.  TNH 19-Dec-77
0000     65 ; 00-05 - Return after main prog., using OWN storage.  TNH 6-Jan-78.
0000     66 ; 00-08 - Change PSECT to LIB$CODE NOSHR.  JMT 25-Jan-78
0000     67 ; 00-09 - Declare PSECT GBL.  TNH 10-Jun-78
0000     68 ; 00-10 - .PSECT LIB$DATA NOPIC.  TNH 28-June-78
0000     69 ; 1-001 - Update version number and copyright notice.  JBS 16-NOV-78
0000     70 ; 1-002 - Remove $SFDEF macro - not needed.  JBS 16-DEC-78
0000     71 ; 1-003 - Add "_" to certain PSECT directives.  (Don't change those
0000     72 ;          documented in release 1.)  JBS 21-DEC-78
0000     73 ; 1-004 - Zero the "Environment Pointer" (R1) before calling
0000     74 ;          initialization routines.  This insures that routines
0000     75 ;          written in PASCAL, PL/I, etc. will work correctly.
0000     76 ;          SBL 7-April-1980
```

```
                     0000     78          .SBTTL  DECLARATIONS
                     0000     79
                     0000     80 ;
                     0000     81 ; INCLUDE FILES:
                     0000     82 ;
                     0000     83 ;
                     0000     84 ; EXTERNAL SYMBOLS:
                     0000     85 ;
                     0000     86          .DSABL  GBL                     ; Force .EXTRN declarations for externals
                     0000     87 ;
                     0000     88 ; MACROS:
                     0000     89 ;
                     0000     90
                     0000     91 ;
                     0000     92 ; PSECT DECLARATIONS:
                     0000     93 ;
                 00000000     94          .PSECT  LIB$INITIALIZDZ NOPIC,USR,CON,REL,GBL,NOSHR,NOEXE,RD,NOWRT,LONG
                     0000     95 INIT_TABLE_BEG:                          ; First entry in table
                 00000000     96          .PSECT  LIB$INITIALIZD_ NOPIC,USR,CON,REL,GBL,NOSHR,NOEXE,RD,NOWRT,LONG
                     0000     97                                          ; Contribution from OTS$STARTUP
                 00000000     98          .PSECT  LIB$INITIALIZE  NOPIC,USR,CON,REL,GBL,NOSHR,NOEXE,RD,NOWRT,LONG
                     0000     99                                          ; Contributions from rest of library and use
                 00000000    100          .PSECT  LIB$INITIALIZE$ NOPIC,USR,CON,REL,GBL,NOSHR,NOEXE,RD,NOWRT,LONG
                     0000    101 INIT_TABLE_END:
            00000000 0000    102          .LONG   0                       ; End of startup list
                     0004    103                                          ; The code for this module
                     0004    104
                     0004    105 ;
                     0004    106 ; EQUATED SYMBOLS:
                     0004    107 ;
                     0004    108 ;
                     0004    109 ; OWN STORAGE:
                     0004    110 ;
                     0004    111
                 00000000    112          .PSECT  _LIB$DATA       PIC,USR,CON,REL,LCL,NOSHR,NOEXE,RD,WRT
                     0000    113
                     0000    114 DISP_PTR:                                ; OWN variable containing
            00000004 0000    115          .BLKL   1                       ; pointer to initialization dispatch table
                     0004    116 SAV_START_ADR:                           ; OWN variable to save original
            00000008 0004    117          .BLKL   1                       ; value of START_ADR_ADR arg list entry
            0000000C 0008    118 SAV_AP:  .BLKL   1                       ; saved value of AP when called
                     000C    119                                          ; needed to find original arg list
                     000C    120                                          ; on co-routine call back.
                     000C    121
                 00000000    122          .PSECT  _LIB$CODE_NOSHR PIC,USR,CON,REL,LCL,SHR,EXE,RD,NOWRT
```

```
                                0000   124          .SBTTL  LIB$INITIALIZE - Library initialization procedure dispatcher
                                0000   125
                                0000   126 ;++
                                0000   127 ; FUNCTIONAL DESCRIPTION:
                                0000   128 ;
                                0000   129 ;       LIB$INITIALIZE is a library procedure for initializing
                                0000   130 ;       library and user procedures at run-time which cannot
                                0000   131 ;       be done statically at compile-time or link-time.
                                0000   132 ;       LIB$INITIALIZE is called before the main program
                                0000   133 ;       if the GLOBAL symbol LIB$INITIALIZE is defined.  User procedures
                                0000   134 ;       and library procedures need only declare LIB$INITIALIZE
                                0000   135 ;       as an EXTERNAL (the compiler generates EXTERNAL OTS$LINKAGE
                                0000   136 ;       which generates the EXTERNAL LIB$INITIALIZE).
                                0000   137 ;       LIB$INITIALIZE calls every entrypoint contributed to
                                0000   138 ;       PSECT LIB$INITIALIZE.
                                0000   139 ;       Note:  this procedure is not re-entrant.  It can only be called
                                0000   140 ;       once per image activation as currently coded.   If multiple
                                0000   141 ;       main programs exist in an image, all but the first should be called
                                0000   142 ;       directly, rather than calling through the image startup list
                                0000   143 ;       setup by the linker.
                                0000   144 ;
                                0000   145 ; CALLING SEQUENCE:
                                0000   146 ;
                                0000   147 ;       Status.wlc.v = LIB$INITIALIZE (start_adr_adr.ma.r, cli_co_rout.flc.r, ...)
                                0000   148
                                0000   149 ;
                                0000   150 ; INPUT PARAMETERS:
                                0000   151 ;
                    00000004    0000   152          START_ADR_ADR = 4                ; Adr. of the entry
                                0000   153                                          ; in the image startup vector
                                0000   154                                          ; which is used to perform the call.
                                0000   155
                                0000   156 ;
                                0000   157 ; IMPLICIT INPUTS:
                                0000   158 ;       NONE
                                0000   159 ;
                                0000   160 ; OUTPUT PARAMETERS:
                                0000   161 ;       NONE
                                0000   162 ;
                                0000   163 ; IMPLICIT OUTPUTS:
                                0000   164 ;       NONE
                                0000   165 ;
                                0000   166 ; COMPLETION CODES:
                                0000   167 ;
                                0000   168 ;       The completion code returned from the main program.
                                0000   169 ;
                                0000   170 ; SIDE EFFECTS:
                                0000   171 ;       Violates VAX-11 Procedure Calling standard by modifying arg list entry START
                                0000   172 ;       Increments it by 4 to point to main program starting address.
                                0000   173 ;       This violation is acceptable, since LIB$INITIALIZE only called
                                0000   174 ;       by CLI and debugger.
                                0000   175 ;--
                                0000   176
                                0000   177
                                0000   178
```

```
                                                H 16
                          4000  0000   180          .ENTRY  LIB$INITIALIZE, ^M<IV>  ; entry point and entry mask
                                0002   181                                          ; enable integer overflow
          00000008'EF    5C    D0  0002   182          MOVL    AP, SAV_AP              ; save AP for co-routine restore
     00000004'EF    04 AC    D0  0009   183          MOVL    START_ADR_ADR(AP), -    ; save pointer to startup list in OWN
                                0011   184                  SAV_START_ADR           ; storage (SAV_START_ADR)
               04 AC   53'AF    DE  0011   185          MOVAL   B^CO_ROUT, -            ; change arg list entry in case
                                0016   186                  START_ADR_ADR(AP)       ; an initialization procedure is called.
00000000'EF    00000000'EF    DE  0016   187          MOVAL   INIT_TABLE_BEG, -       ; initialize OWN storage (DISP_PTR)
                                0021   188                  DISP_PTR                ; to first address of user/library
                                0021   189
                                0021   190 ;+
                                0021   191 ; Loop to scan table of user and library initialization procedure
                                0021   192 ; starting addresses in PSECT LIB$INITIALIZDZ, LIB$INITIALIZD_,
                                0021   193 ; LIB$INITIALIZE, and LIB$INITIALIZE$
                                0021   194 ;-
                                0021   195
          50    00000000'FF    D0  0021   196 LOOP:   MOVL    @DISP_PTR, R0           ; R0 = next user/library initialization
                                0028   197                                          ; procedure address
                       16    13  0028   198          BEQL    CALL_MAIN               ; zero means end of list
                       51    D4  002A   199          CLRL    R1                      ; Clear environment pointer
                    60    6C    FA  002C   200          CALLG   (AP), (R0)              ; call next user/library init proc.
                00000000'FF    D5  002F   201          TSTL    @DISP_PTR               ; test if dispatch address is zero
                       1B    13  0035   202          BEQL    RET                     ; if yes, this must be return from
                                0037   203                                          ; user initialization proc. which had
                                0037   204                                          ; done co-routine call-back.
                                0037   205                                          ; There fore this is return from main
                                0037   206                                          ; program, so return to caller (of LIB$INIT
                                0037   207                                          ; caller or initializtion procedure)
                                0037   208                                          ; may loop thru here for each init proc whic
                                0037   209
                                0037   210 ;+
                                0037   211 ; Here from co-routine call-back from an initialization routine
                                0037   212 ;-
                                0037   213
                                0037   214 DISPATCH_CONT:                           ; continue dispatching to initialization pro
          00000000'EF    04    C0  0037   215          ADDL    #4, DISP_PTR            ; step dispatch pointer to next
                                003E   216                                          ; init. proc. address
                       E1    11  003E   217          BRB     LOOP                    ; loop
                                0040   218
                                0040   219 ;+
                                0040   220 ; Here when entire initialization list has been scanned
                                0040   221 ; Call main program
                                0040   222 ;-
                                0040   223
                                0040   224 CALL_MAIN:                             ; here to call main program
04 AC     00000004'EF    04    C1  0040   225          ADDL3   #4, SAV_START_ADR, -    ; Restore initial argument and step
                                0049   226                  START_ADR_ADR(AP)       ; to next longword in image startup vector.
                  50    04 BC    D0  0049   227          MOVL    @START_ADR_ADR(AP), R0  ; R0 = next start address (main program)
                       51    D4  004D   228          CLRL    R1                      ; Clear environment pointer
                    60    6C    FA  004F   229          CALLG   (AP), (R0)              ; call main program or main procedure
                                04  0052   230 RET:    RET                             ; R0 = status returned by main program
```

```
                                                I 16
                                0053   232
                                0053   233 ;+
                                0053   234 ; Co-routine - may be called by library or user initialization procedure.
                                0053   235 ; Set AP to value it had when LIB$INITIALIZE originally called.
                                0053   236 ; This lets LIB$INITIALIZE re-use the same argument list
                                0053   237 ;-
                                0053   238
                                0053   239 CO_ROUT:
                           4000 0053   240         .WORD   ^M<IV>                  ; no registers used.
      5C  00000008'EF       D0 0055   241         MOVL    SAV_AP, AP              ; restore AP to original arg list
                                005C   242                                         ; when LIB$INITIALIZE was called.
                        D9  11 005C   243         BRB     DISPATCH_CONT           ; go continue dispatch loop
                                005E   244
                                005E   245         .END                            ; End of module LIB$INITIALIZE
```

```
CALL_MAIN           00000040 R      06
CO_ROUT             00000053 R      06
DISPATCH_CONT       00000037 R      06
DISP_PTR            00000000 R      05
INIT_TABLE_BEG      00000000 R      01
INIT_TABLE_END      00000000 R      04
LIB$INITIALIZE      00000000 RG     06
LOOP                00000021 R      06
RET                 00000052 R      06
SAV_AP              00000008 R      05
SAV_START_ADR       00000004 R      05
START_ADR_ADR   =   00000004
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
LIB$INITIALIZDZ	00000000 (0.)	01 (1.)	NOPIC USR CON REL GBL NOSHR NOEXE RD NOWRT NOVEC LONG
LIB$INITIALIZD_	00000000 (0.)	02 (2.)	NOPIC USR CON REL GBL NOSHR NOEXE RD NOWRT NOVEC LONG
LIB$INITIALIZE	00000000 (0.)	03 (3.)	NOPIC USR CON REL GBL NOSHR NOEXE RD NOWRT NOVEC LONG
LIB$INITIALIZE$	00000004 (4.)	04 (4.)	NOPIC USR CON REL GBL NOSHR NOEXE RD NOWRT NOVEC LONG
_LIB$DATA	0000000C (12.)	05 (5.)	PIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC BYTE
_LIB$CODE_NOSHR	0000005E (94.)	06 (6.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.03	00:00:02.66
Command processing	109	00:00:00.30	00:00:03.67
Pass 1	67	00:00:00.35	00:00:02.97
Symbol table sort	0	00:00:00.00	00:00:00.01
Pass 2	59	00:00:00.28	00:00:01.36
Symbol table output	2	00:00:00.02	00:00:00.02
Psect synopsis output	4	00:00:00.01	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	272	00:00:01.00	00:00:10.71

```
The working set limit was 750 pages.
2647 byies (6 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 12 non-local and 0 local symbols.
245 source lines were read in Pass 1, producing 20 object records in Pass 2.
0 pages of virtual memory were used to define 0 macros.
```

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	0

```
0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:LIBINITIA/OBJ=OBJ$:LIBINITIA MSRC$:LIBINITIA/UPDATE=(ENH$:LIBINITIA)
```

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

LIBFLTUND
LIS

LIBGETSYI
LIS

LIBICHAR
LIS

LIBINITIA
LIS

LIBFIXUPF
LIS

LIBGETFOR
LIS

LIBGETINP
LIS

LIBINISHR
LIS

LIBGETDVI
LIS

LIBGETOPC
LIS

LIBFNDIMG
LIS

LIBGETMSG
LIS

LIBINDEX
LIS

LIBINSQHI
LIS

LIBGETJPI
LIS

LIBGETTAB
LIS